View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by INRIA a CCSD electronic archive server

Structured Feature Selection of Continuous Dynamical Systems for Aircraft Dynamics Identification

Cédric Rommel, Frédéric Bonnans, Baptiste Gregorutti, Pierre Martinon

To cite this version:

Cédric Rommel, Frédéric Bonnans, Baptiste Gregorutti, Pierre Martinon. Structured Feature Selection of Continuous Dynamical Systems for Aircraft Dynamics Identification. 2018. hal-01965959

HAL Id: hal-01965959

https://hal.archives-ouvertes.fr/hal-01965959

Preprint submitted on 27 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Structured Feature Selection of Continuous Dynamical Systems for Aircraft Dynamics Identification

C. Rommel[1,2,3], J. F. Bonnans[1,2], B. Gregorutti[3] and P. Martinon[1,2]
INRIA[1], CMAP[2], Safety Line[3]
cedric@ava.me · frederic.bonnans@inria.fr
baptiste.gregorutti@safety-line.fr · pierre.martinon@inria.fr

Abstract

This paper addresses the problem of identifying structured nonlinear dynamical systems, with the goal of using the learned dynamics in model-based reinforcement learning problems. We present in this setting a new class of scalable multi-task estimators which promote sparsity, while preserving the dynamics structure and leveraging available physical insight. An implementation leading to consistent feature selection is suggested, allowing to obtain accurate models. An additional regularizer is also proposed to help in recovering realistic hidden representations of the dynamics. We illustrate our method by applying it to an aircraft trajectory optimization problem. Our numerical results based on real flight data from 25 medium haul aircraft, totaling 8 millions observations, show that our approach is competitive with existing methods for this type of application.

Keywords: model-based reinforcement learning, optimal control, system identification, structured feature selection, strong correlations

1 Introduction

Using past data to learn how to control a system efficiently with relation to some predefined criterion is one of the main objectives of control theory. More recently, the machine learning community has independently addressed this class of problems, which resulted in the emergence of the reinforcement learning subfield. The practical applications of the techniques developped within both these communities span from ensuring that an air-conditioning system keeps a room at a specific temperature, to mastering Go [Silver et al., 2017] and safely controlling self-driving cars, aircrafts and rockets.

While reinforcement learning is predominantly *model-free* and tries to find the solutions solely by analyzing the data, control theory is usually based on the use of predefined models of the system dynamics. When the dynamics are unknown, the standard approach adopted by control engineers is to split the problem in two steps: (i) first the dynamics are learned from previously measured data, (ii) then the optimization problem cast using the estimated model

is solved. The first step is known in the control literature as *system identification* [Ljung, 1987] and relates to well-known supervised learning tools, such as maximum likelihood parameter estimation. This illustrates the similarities that exist between the control approaches and the reinforcement learning approaches, which explains why the former has recently been called *model-based reinforcement learning* by some authors [Recht, 2018]. We believe that this class of methods are of particular interest for the reinforcement learning community as they have been shown in recent numerical results [Dean et al., 2017, Recht, 2018] to have significant better sample complexity than classical model-free approaches in the context of linear and nonlinear *continuous* systems.

This paper addresses the study of model-based approaches for solving continuous optimal control problems with unknown dynamics. More precisely, our main focus lies on the system identification of structured nonlinear dynamical systems. Besides briefly presenting this type of problems and existing techniques to solve them, the main contribution of this article is to propose a new grey-box identification approach which is particularly suitable when large amounts of data are available. We show that our method allows to find descriptions which are flexible enough to cover many relevant nonlinear phenomena, while making use of physical insight to improve its generalization. This is achieved by making use of a regularized multi-task regression formulation and structured feature selection, which lead to dynamics models that are light, interpretable and fast to evaluate.

The remaining of the article is organized as follows. In Section 2 we present with more details the model-based approach which motivates the need for accurate system identification techniques, while Section 3 is a brief summary of the existing well-established methods. We define in Section 4 the broad class of dynamical systems said to be *structured*, and explain in Section 5 how the identification of most of these systems can be cast as linear regression problems. Section 6 is devoted to motivating and describing a first multi-task structured feature selection technique suited for this type of problem, which we call *block-sparse lasso*. Because of the undesirable behaviors of this algorithm when the model features are strongly correlated, two adaptations making use of bootstrap stabilization and generalized Tikhonov regularization are proposed in Section 7. All statistical models suggested are shown to be equivalent to surrogate lasso problems, which can be efficiently solved by well-known existing optimization algorithm. After presenting an application to aircraft trajectory optimization in Section 8, experiments using a real data set of 10 471 recorded flights are carried out in Section 9 to evaluate the performance of our approach and compare it to other existing techniques.

2 Optimal Control with Unknown Dynamics

We consider a dynamical system whose state $\boldsymbol{x} \in \mathcal{X} \subset W^{1,\infty}(0, t_f; \mathbb{R}^{d_x})$ is a continuous function supposed to be controlled by some functional inputs $\boldsymbol{u} \in \mathcal{U} \subset L^{\infty}(0, t_f; \mathbb{R}^{d_u})$ through a system of ODEs often called *dynamics*:

$$\dot{\boldsymbol{x}}(t) = g(\boldsymbol{u}(t), \boldsymbol{x}(t)). \tag{1}$$

In an optimal control problem, one will seek the controls which minimize over some time horizon t_f a certain cost function C, called *running cost*, under a few

constraints, which include the dynamics (1):

$$\min_{(\boldsymbol{x},\boldsymbol{u})} \int_0^{t_f} C(t, \boldsymbol{u}(t), \boldsymbol{x}(t))dt,$$
$$\text{s.t.} \begin{cases} \dot{\boldsymbol{x}}(t) = g(\boldsymbol{u}(t), \boldsymbol{x}(t)), \quad \text{for a.e. } t \in [0, t_f], \\ \boldsymbol{u}(t) \in U_{ad}, \quad \boldsymbol{x}(t) \in X_{ad}, \quad \text{for a.e. } t \in [0, t_f], \\ \Phi(\boldsymbol{x}(0), \boldsymbol{x}(t_f)) \in K_\Phi, \\ c(t, \boldsymbol{u}(t), \boldsymbol{x}(t)) \leq 0, \quad \text{for a.e. } t \in [0, t_f]. \end{cases} \tag{2}$$

All mappings in (2), i.e. the *running cost* C, the *dynamics* function g, the *initial-final state constraint* function Φ and the *path constraint* function c are assumed to be continuously differentiable. The sets X_{ad} and U_{ad} are assumed to be closed subsets of $\mathbb{R}^{d_x}$ and $\mathbb{R}^{d_u}$ respectively, and K_Φ is a nonempty closed convex subset of $\mathbb{R}^{n_\Phi}$, $n_\Phi \in \mathbb{N}^*$.

Although the dynamics function g is a key element of problem (2), in most real world application cases, it is unknown. Indeed, while physical models of g often arise from the use of first-principles and domain-specific knowledge, most of the times they depend on unknown parameters $\boldsymbol{\theta}$:

$$\dot{\boldsymbol{x}}(t) = g(\boldsymbol{u}(t), \boldsymbol{x}(t), \boldsymbol{\theta}). \tag{3}$$

When noisy observations of the system state and control variables are available, one may use them to learn $\boldsymbol{\theta}$. Supposing that the estimated dynamics $\hat{g} = g(\cdot, \cdot, \hat{\boldsymbol{\theta}})$ are close to the real dynamics g, model-based approaches assume that good approximate solutions of (2) can be obtained by solving the surrogate problem

$$\min_{(\boldsymbol{x},\boldsymbol{u})} \int_0^{t_f} C(t, \boldsymbol{u}(t), \boldsymbol{x}(t))dt,$$
$$\text{s.t.} \begin{cases} \dot{\boldsymbol{x}}(t) = \hat{g}(\boldsymbol{u}(t), \boldsymbol{x}(t)), \quad \text{for a.e. } t \in [0, t_f], \\ \boldsymbol{u}(t) \in U_{ad}, \quad \boldsymbol{x}(t) \in X_{ad}, \quad \text{for a.e. } t \in [0, t_f], \\ \Phi(\boldsymbol{x}(0), \boldsymbol{x}(t_f)) \in K_\Phi, \\ c(t, \boldsymbol{u}(t), \boldsymbol{x}(t)) \leq 0, \quad \text{for all } t \in [0, t_f]. \end{cases} \tag{4}$$

This motivates the need for accurate system identification techniques to learn potentially nonlinear dynamics.

3 System Identification State-Of-the-Art

The identification of linear systems (i.e. linear with relation to the states and controls) have been extensively studied in the classical control literature and is a mature field. Indeed, many well-know methods based on Laplace and Fourier transform allow to efficiently estimate the parameters with good theoretical properties [see e.g. Ljung, 1987, chapter 4]). However, most real world phenomena which we try to model using dynamical systems are nonlinear, and existing techniques to identify such models are more demanding and not as mature yet. This explains why the study of nonlinear dynamics is probably the most active area of research in system identification today [see e.g. Ljung, 2010].

Most well-established methods for nonlinear system identification are based on solving the system of ODEs (3) several times. Suppose that we are in the *full*

observation framework (meaning that the states $\boldsymbol{x}$ are directly observable) and that the available data is made of n discrete measurements of m trajectories

$$\Big\{(\boldsymbol{u}^r(t_i), \boldsymbol{x}^r(t_i))\Big\}_{\substack{1\leq r\leq m\\ 1\leq i\leq n}}.$$

For each trajectory $\{(\boldsymbol{u}^r(t_i), \boldsymbol{x}^r(t_i))\}_{i=1}^n$, a control function $\hat{\boldsymbol{u}}^r : [t_1, t_n] \to \mathbb{R}^{d_u}$ can be approximated using the discrete observations $\{\boldsymbol{u}^r(t_i)\}_{i=1}^n$, using for example splines or piecewise linear functions. For a given trial of the parameters $\boldsymbol{\theta}_k \in \mathbb{R}^p$ at iteration k, the observed states $(\boldsymbol{x}^r(t_i))_{i=1}^n$ can be resimulated using the continuous control function by solving the following *initial-value problem* or *simulation* problem:

$$\begin{cases} \dot{\hat{\boldsymbol{x}}}(t) = g(\hat{\boldsymbol{u}}^r(t), \boldsymbol{x}(t), \boldsymbol{\theta}_k), \quad \text{for all } t \in [t_1, t_n], \\ \boldsymbol{x}(t_1) = \boldsymbol{x}^r(t_1). \end{cases} \tag{5}$$

As (5) usually does not have an analytical solution, numerical integration algorithms, such as Runge-Kutta schemes, are used to solve discrete approximations of it. We denote by $\hat{\boldsymbol{x}}^r(\boldsymbol{\theta}_k) = (\hat{\boldsymbol{x}}^r(\boldsymbol{\theta}_k, t_i))_{i=1}^n \in \mathbb{R}^n$ the approximate solution of the simulation problem (5) and $\hat{\boldsymbol{x}}(\boldsymbol{\theta}_k) = (\hat{\boldsymbol{x}}^r(\boldsymbol{\theta}_k))_{r=1}^m \in \mathbb{R}^{mn}$. For some convex *loss function* $\mathcal{L} : \mathbb{R}^2 \to \mathbb{R}_+$, standard identification methods [see e.g. Betts, 2010, chapter 5] estimate the parameters $\boldsymbol{\theta}$ by minimizing the simulation error using standard optimization algorithms:

$$\begin{aligned} &\min_{\boldsymbol{\theta}, \hat{\boldsymbol{x}}} \sum_{r=1}^{m} \sum_{i=1}^{n} \mathcal{L}(\hat{\boldsymbol{x}}^r(\boldsymbol{\theta}, t_i), \boldsymbol{x}^r(t_i)) \\ &\text{such that } \hat{\boldsymbol{x}}^r(\boldsymbol{\theta}) \text{ satisfies (5) for } r = 1, \dots, m. \end{aligned} \tag{6}$$

This class of techniques are sometimes called *output-error methods* [Jategaonkar, 2006, Maine and Iliff, 1985]. In order to account for stochastic process errors in the dynamics, some variations of these methods replace the deterministic numerical integration schemes used to solve (5) by state estimators, such as Kalman filters [Peyada et al., 2008] or Neural Networks [Peyada and Ghosh, 2009]. These approaches are sometimes called *filter-error methods* [Klein and Morelli, 2006].

Because these methods are based on the simulation of every trajectory of the training set for each parameter update, they are known to be computationally intensive. For this reason, these techniques are not applicable to situations with many measured trajectories. Indeed, the dimension of the optimization problem (6) increases linearly with the number of trajectories m, as it is equal to $p + m \times n \times (d_x + d_u)$.

In this paper we focus on another classical class of techniques, called *equation-error methods*, which are known to be more scalable, although they are believed to be less accurate than the previously explained methods. In these approaches, the dynamic nature of the data is ignored and the observations of different trajectories are concatenated so as to form an unordered set of observations $\{(\boldsymbol{u}_i, \boldsymbol{x}_i)\}_{i=1}^N$, where $N = nm$. The dynamics equation (3) is then rearranged so as to cast a regression problem. The input variables $X_i = (\boldsymbol{u}_i, \boldsymbol{x}_i, \dot{\boldsymbol{x}}_i) \in \mathbb{R}^{2d_x+d_u}$ are the controls, states and states derivatives. The output variables $Y_i = \Psi(\boldsymbol{u}_i, \boldsymbol{x}_i, \dot{\boldsymbol{x}}_i) \in \mathbb{R}^{d_x}$ depend on these same variables through some mapping Ψ derived from the dynamics expression and a random variable $\varepsilon_i \in \mathbb{R}^{d_x}$

accounts for the identification and process errors. Finally, the regression problem obtained writes as

$$Y_i = G(X_i, \boldsymbol{\theta}) + \varepsilon_i, \qquad i = 1, \ldots, N. \tag{7}$$

A possible choice here for Ψ and G is to take $\Psi(\boldsymbol{u}, \boldsymbol{x}, \dot{\boldsymbol{x}}) = \dot{\boldsymbol{x}}$ and $G(X, \boldsymbol{\theta}) = g(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta})$. Depending on the problem, better formulations can for instance isolate elements not depending on the parameters $\boldsymbol{\theta}$ in the left-hand side. For example, assume without loss of generality that $d_x = 1$ and suppose that the dynamics have the following general shape

$$g(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) = \left(\frac{g_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) + g_2(\boldsymbol{u}, \boldsymbol{x})}{\tilde{g}_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) + \tilde{g}_2(\boldsymbol{u}, \boldsymbol{x})} \right) \tilde{g}_3(\boldsymbol{u}, \boldsymbol{x}).$$

In this case, we can rearrange equation (3) as follows

$$\frac{\dot{\boldsymbol{x}}}{\tilde{g}_3(\boldsymbol{u}, \boldsymbol{x})} \left(\tilde{g}_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) + \tilde{g}_2(\boldsymbol{u}, \boldsymbol{x}) \right) = g_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) + g_2(\boldsymbol{u}, \boldsymbol{x}) \tag{8}$$

$$\frac{\dot{\boldsymbol{x}}}{\tilde{g}_3(\boldsymbol{u}, \boldsymbol{x})} \tilde{g}_2(\boldsymbol{u}, \boldsymbol{x}) - g_2(\boldsymbol{u}, \boldsymbol{x}) = g_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) - \frac{\dot{\boldsymbol{x}}}{\tilde{g}_3(\boldsymbol{u}, \boldsymbol{x})} \tilde{g}_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}), \tag{9}$$

leading to

$$\Psi(\boldsymbol{u}, \boldsymbol{x}, \dot{\boldsymbol{x}}) = \frac{\dot{\boldsymbol{x}}}{\tilde{g}_3(\boldsymbol{u}, \boldsymbol{x})} \tilde{g}_2(\boldsymbol{u}, \boldsymbol{x}) - g_2(\boldsymbol{u}, \boldsymbol{x}) \tag{10}$$

$$G(X, \boldsymbol{\theta}) = g_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}) - \frac{\dot{\boldsymbol{x}}}{\tilde{g}_3(\boldsymbol{u}, \boldsymbol{x})} \tilde{g}_1(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}). \tag{11}$$

A practical application where this type of formulation is useful is given in Section 8.

Remark Note that the *equation-error methods* assume that observations of the states derivatives $\dot{\boldsymbol{x}}$ are available. When this is not the case, the measured trajectories can be smoothed using a basis function expansion (usually splines), and the analytical form of the smoothed signals can be used to compute an approximation of the derivatives. This subclass of approaches are sometimes called *collocation methods* [Ramsay et al., 2007, Varah, 1982].

4 Learning Structured Dynamics in a Multi-task Setting

In many practical cases, dynamical systems present some type of coupling structure between its state variables. This translates into elements which are shared by the differential equations governing the system dynamics. This structure can be seen for example in the FitzHugh-Nagumo model of the behaviour of spike potentials in the giant squid neurons [FitzHugh, 1961, Nagumo et al., 1962]:

$$\begin{cases} \dot{V} = \boldsymbol{\theta}_1 \left(V - \dfrac{V^3}{3} + R \right), \\ \dot{R} = -\dfrac{1}{\boldsymbol{\theta}_1} (V - \boldsymbol{\theta}_2 + \boldsymbol{\theta}_3 R), \end{cases} \tag{12}$$

where V denotes the voltage across an axon membrane and R corresponds to the outward currents. We see indeed that the parameter $\boldsymbol{\theta}_1$ in (12) is shared by both equations. Other examples are the susceptible-infectious-recovered (SIR) models [Anderson and May, 1992] describing the dynamics of a population attacked by an infectious disease, such as:

$$\begin{cases} \dot{S} = \boldsymbol{\theta}_1 - (\boldsymbol{\theta}_2 I + \boldsymbol{\theta}_3) S, \\ \dot{I} = \boldsymbol{\theta}_2 I S - (\boldsymbol{\theta}_4 + \boldsymbol{\theta}_3) I, \\ \dot{R} = \boldsymbol{\theta}_4 I - \boldsymbol{\theta}_3 R. \end{cases} \tag{13}$$

In this model, S, I and R denote respectively the number of susceptible, infectious and recovered (or immune) people. Here again, the model parameter $\boldsymbol{\theta}_3$ is shared by all three equations, and $\boldsymbol{\theta}_4$ appears in the ODEs of both I and R. The dynamics of an aircraft is another interesting example of a coupled dynamical system which is presented with more details in Section 8.1.

From the perspective of the equation-error method, this framework leads regression (7) to take the form of a multi-task regression problem:

$$\begin{cases} Y_1 &= G_{s,1}(X, \boldsymbol{\theta}_s) + G_1(X, \boldsymbol{\theta}_1) + \varepsilon_1, \\ Y_2 &= G_{s,2}(X, \boldsymbol{\theta}_s) + G_2(X, \boldsymbol{\theta}_2) + \varepsilon_2, \\ \vdots & \qquad \vdots \\ Y_{d_x} &= G_{s,d_x}(X, \boldsymbol{\theta}_s) + G_{d_x}(X, \boldsymbol{\theta}_{d_x}) + \varepsilon_{d_x}, \end{cases} \tag{14}$$

where $\boldsymbol{\theta}_s \in \mathbb{R}^{p_s}$ denotes the parameters shared by the differential equations and $(\boldsymbol{\theta}_k)_{k=1}^{d_x} \in \bigotimes_{k=1}^{d_x} \mathbb{R}^{p_k}$ denote the parameters which are specific to each equation. Solving all d_x regressions from (14) simultaneously may seem quite natural in the context of modern system identification, since it enforces the agreement of all components of the estimated dynamics by insuring they share the same coupling parameters. Moreover, from a statistical learning viewpoint, it is well-known that learning multiple tasks simultaneously may increase the predictive accuracy when some coupling exist, as observed in many other multi-task learning applications [Caruana, 1997, Evgeniou et al., 2005]. In the specific case of system identification, it was indeed demonstrated through experiments presented in Rommel et al. [2017], that leveraging the coupling structure of the dynamical system can improve the predictive accuracy. As explained with more details in the following sections, we intend to push this line of thought further in this study by proposing a structured multi-task feature selection procedure.

5 Linearity in Parameters

It is not uncommon to deal with nonlinear systems which are linear on their parameters, such as the SIR model (13). Moreover, this subclass of systems is broader than it seems as it has been proven in Ljung and Glad [1994] that *any globally identifiable dynamical system can be rearranged using Ritt's algorithm [Ritt, 1950] into a system which is linear on its parameters.* This is all-the-more interesting as this property naturally facilitates the theoretical and computational aspects of the identification, as it brings (14) to become a multi-task

linear regression problem:

$$\begin{cases} Y_1 &= \varphi_{s,1}(X)^\top \boldsymbol{\theta}_s + \varphi_1(X)^\top \boldsymbol{\theta}_1 + \varepsilon_1 \\ Y_2 &= \varphi_{s,2}(X)^\top \boldsymbol{\theta}_s + \varphi_2(X)^\top \boldsymbol{\theta}_2 + \varepsilon_2 \\ \vdots & \quad \vdots \\ Y_{d_x} &= \varphi_{s,d_x}(X)^\top \boldsymbol{\theta}_s + \varphi_{d_x}(X)^\top \boldsymbol{\theta}_{d_x} + \varepsilon_{d_x}, \end{cases} \tag{15}$$

where $\varphi_{s,k}$ and φ_k, $k = 1, \ldots, d_x$, are predefined shared and specific feature maps from $\mathbb{R}^{2d_x+d_u}$ to $\mathbb{R}^{p_s}$ and $\mathbb{R}^{p_k}$ respectively. Assuming that X and Y are random variables and that we have access to N i.i.d observations of them $\{(X_i, Y_i)\}_{i=1}^N$, this formulation leads to the following optimization problem

$$\min_{\boldsymbol{\theta}} \sum_{k=1}^{d_x} \sum_{i=1}^{N} \mathcal{L}\left(Y_{i,k}, \varphi_{s,k}(X_i)^\top \boldsymbol{\theta}_s + \varphi_k(X_i)^\top \boldsymbol{\theta}_k\right) + \mathcal{R}_\lambda(\boldsymbol{\theta}), \tag{16}$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_s, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{d_x})$ is the vector of length $p = p_s + \sum_{k=1}^{d_x} p_k$ containing all parameters, $\mathcal{L} : \mathbb{R}^2 \to \mathbb{R}_+$ is some loss function, $\mathcal{R}_\lambda : \mathbb{R}^p \to \mathbb{R}_+$ is a regularizer depending on some parameter λ.

6 Block-Sparse Feature Selection

Sparse models are attractive in many application domains since they lend themselves particularly well to interpretation and are lighter. This is particularly interesting in the context of model-based reinforcement learning, as identified models are meant to be plugged in optimal control solvers, which often require the dynamics to be fast to compute. Moreover, promoting sparsity reduces the statistical model complexity and has been proven to enhance the predictive accuracy, which can help to make the *equation-error method* competitive with more well-established approaches regarding this aspect.

One natural way of trying to select a few number of relevant features in the context of our structured identification problem (16) is by choosing $\mathcal{L}$ to be the squared loss and $\mathcal{R}_\lambda$ to be the L^1 norm:

$$\min_{\boldsymbol{\theta}} \sum_{k=1}^{d_x} \left(\sum_{i=1}^{N} \left(Y_{i,k} - \varphi_{s,k}(X_i)^\top \boldsymbol{\theta}_s - \varphi_k(X_i)^\top \boldsymbol{\theta}_k\right)^2 + \lambda_k \|\boldsymbol{\theta}_k\|_1 \right) + \lambda_s \|\boldsymbol{\theta}_s\|_1. \tag{17}$$

Indeed, the sparsity-inducing properties of the L^1 norm are well-known since the original paper on the lasso by Tibshirani [1994], which considered a (single-task) least-squares linear regression setting. Since then, this same penalty has been successfully applied to many other loss functions such as Cox regression [Tibshirani, 1997] and logistic regression [Krishnapuram et al., 2005, Lokhorst, 1999]. Despite the great usefulness of the original lasso penalty for inducing sparsity in statistical models, it does not encode any structural information based on prior knowledge about the problem. The *group lasso* [Yuan and Lin, 2005] was probably one of the first of its extensions to cope with this limitation. It selects entire groups of variables using the $L^{2,1}$ penalty, which is the sum (i.e. L^1-norm) of the L^2-norms of subsets forming a partition of the features. A variation of this method, coined the *sparse-group lasso* [Friedman et al., 2010], promotes sparsity not only between predefined groups of variables, but also

within these groups. This is achieved by adding to the *group lasso* criterion a L^1 penalty over the whole parameters vector. Moreover, the *multi-task lasso* [Argyriou et al., 2008, Obozinski et al., 2006] assumes that all tasks in a multiple outputs regression problem share the same features and also uses the structuring properties of the $L^{2,1}$-norm to select them.

Despite the multi-task setting of (17), our approach differs from the original *multi-task lasso* method, as we do not assume that all tasks share the same features. It rather relates conceptually to the *sparse-group lasso*, since it considers a partition of the problem features into groups and induces sparsity within these groups. However, unlike the sparse-group lasso, an $L^{2,1}$ penalty is not used to define the groups of variables in our case, as the group sparsity pattern is supposed to be known a priori. The variables partitioning structure is instead encoded directly in the loss function part of problem (17), still allowing to use the underlying couplings during the variables selection.

Another important property of our estimator, summarized in the following proposition, is that problem (17) is equivalent to a single-task lasso-type optimization problem, thus enjoying the computational advantage of the lasso.

Proposition 6.1 *Given a data set $\{(X_i, Y_i)\}_{i=1}^N$ and a sequence of coupling $(\varphi_{s,k})_{k=1}^{d_x}$ and specific feature maps $(\varphi_k)_{k=1}^{d_x}$, define for every $k = 1, \ldots, d_x$*

$$B_k(X_i) = \left(\varphi_{s,k}(X_i)^\top, \underbrace{0, \ldots, 0}_{\sum_{i=1}^{k-1} p_i}, \frac{\lambda_s}{\lambda_k} \varphi_k(X_i)^\top, \underbrace{0, \ldots, 0}_{\sum_{i=k+1}^{d_x} p_i} \right) \in \mathbb{R}^p, \tag{18}$$

and

$$B(X_i) = (B_1(X_i)^\top, \ldots, B_{d_x}(X_i)^\top)^\top, \tag{19}$$

which is a $d_x \times p$ matrix. Consider the following surrogate single-task lasso problem

$$\min_\beta \sum_{i=1}^N \|Y_i - B(X_i)\beta\|_2^2 + \lambda_s \|\beta\|_1, \tag{20}$$

and denote its solution by $\hat{\beta} = (\hat{\beta}_s, \hat{\beta}_1, \ldots, \hat{\beta}_{d_x})$, where $\hat{\beta}_s$ corresponds to the first p_s components, $\hat{\beta}_1$ to the following p_1 components and so on. Then, the solution to (17) *becomes*

$$\hat{\boldsymbol{\theta}}_s = \hat{\beta}_s, \quad \textit{and} \quad \hat{\boldsymbol{\theta}}_k = \frac{\lambda_s}{\lambda_k} \hat{\beta}_k, \quad \textit{for } k = 1, \ldots, d_x.$$

This property is really important from a practical point of view as many efficient algorithms exist to solve the surrogate lasso problem (20), such as the LARS [Efron et al., 2004]. Also note that the features matrix $B(X_i)$ in (20) has a block structure, where the only nonzero elements are in the first p_s columns and in the following diagonal blocks of size $1 \times p_k$, $k = 1, \ldots, d_x$. This explains why we call our method *block-sparse lasso* in the remaining of the article. The structuring nature of the loss function announced earlier is also more explicit in formulation (20).

7 Dealing with Strong Correlations

One of the main obstacles that may arise when trying to use the block-sparse lasso for system identification problems are strong correlations. Indeed, as explained in sections 4 and 5, the model features $(\varphi_{s,k}(X), \varphi_k(X))_{k=1}^{d_x}$ emerge from first-principles governing the dynamical system, and may be highly correlated to each other. These high correlations may exist between features $\varphi_k^j(X)$ and $\varphi_k^i(X)$ of a same group $\varphi_k(X) = (\varphi_k^j(X))_{j=1}^{p_k}$, in which case we call them *intra-group correlations*. They may also concern different groups, such as the features of $\varphi_{s,j}(X)$ and $\varphi_k(X)$. These are called *inter-group correlations* hereafter. Both types of correlation will have different negative implications for the application of block-sparse lasso. In this section we explain these implications and suggest ways of circumventing them, while an applicative example is presented in the following section.

7.1 Intra-group Correlations

We showed in proposition 6.1 that the block-sparse lasso is equivalent to a lasso problem. It is well-known that the lasso delivers inconsistent selections when some of the variables are highly correlated [see for example van de Geer, 2010, Zhao and Yu, 2006]. As the block-sparse lasso carries its selection within the features groups, its performances will mainly be impacted by *intra-group correlations*. Indeed, in this case, the variable selection is very sensitive to the training data used and makes a random choice among a couple of correlated features. Many variations of the lasso have since been proposed to cope with this limitation, such as the *adaptive lasso* [Zou, 2006] and *stability selection* [Meinshausen and Bühlmann, 2010]. We suggest to stabilize our feature selection using the *bolasso* algorithm [Bach, 2008], presented hereafter.

Assume that the feature selection is being carried among a set of variables which include those having generated the data. It was shown in Bach [2008] that, under mild assumptions, the lasso *support* (set of selected variables) always includes the correct features, even with a strongly correlated design. Hence, he suggests to perform the lasso repeatedly over several *bootstrap replications* of the initial data set (i.e. samples of size N drawn with replacement using a uniform distribution over the training data). The selected variables J are then given by the intersection of the supports over all lasso executions.

Algorithm 1 Bolasso

training data $\mathcal{T} = \{(X_i, Y_i)\}_{i=1}^N$,
Require: number of bootstrap replicates b,
L^1 penalty parameter λ_s,
for $k = 1$ **to** b **do**
  Generate bootstrap sample $\mathcal{T}_k$,
  Compute lasso estimate $\hat{\boldsymbol{\theta}}^k$ from $\mathcal{T}_k$ by solving (20),
  Compute support $J_k = \{j, \hat{\boldsymbol{\theta}}_j^k \neq 0\}$,
end for
Compute the intersection $J = \bigcap_{k=1}^b J_k$,
Compute $\hat{\boldsymbol{\theta}}_J$ by Ordinary Least-Squares with $\{(B(X_i)_J, Y_i)\}_{i=1}^N$.

The procedure is summarized for our particular framework in algorithm 1, where $B(X)_J$ denotes the matrix obtained when the columns whose indexes are contained in J are removed from $B(X) \in \mathbb{R}^{d_x \times p}$. Let $J^* = \{j, \boldsymbol{\theta}_j \neq 0\}$ be the sparsity pattern of $\boldsymbol{\theta}$, having generated the data. Under mild assumptions, this method has been proved to select the correct variables with probability tending to one:

Theorem 7.1 (Bolasso consistency - Bach [2008]) *For $\lambda = \lambda_0 N^{-\frac{1}{2}}$ and $\lambda_0 > 0$, assume that*

(H1) *the cumulant generating functions $\mathbb{E}\left[\exp(s\|X\|_2^2)\right]$ and $\mathbb{E}\left[\exp(s\|Y\|_2^2)\right]$ are finite for some $s > 0$.*

(H2) *the joint matrix of second order moments $Q = \mathbb{E}\left[XX^\top\right] \in \mathbb{R}^{p \times p}$ is invertible.*

(H3) $\mathbb{E}[Y|X] = X \cdot \boldsymbol{\theta}$ *and* $\mathrm{Var}[Y|X] = \sigma^2$ *a.s. for some $\boldsymbol{\theta} \in \mathbb{R}^p$ and $\sigma \in \mathbb{R}_+^*$.*

Then, for any $b > 0$, the probability that algorithm 1 does not exactly select the correct model has the following upper bound:

$$\mathbb{P}[J \neq J^*] \leq bA_1 e^{-A_2 N} + A_3 \frac{\log N}{N^{1/2}} + A_4 \frac{\log b}{b},$$

where $A_1, A_2, A_3, A_4 > 0$.

Indeed, if $\log(b)$ tends to infinity slower than N when N tends to infinity, the bolasso asymptotically selects the correct variables with overwhelming probability. The usefulness of this stabilization algorithm is demonstrated with real data for aircraft dynamics identification in sections 8 and 9.

7.2 Inter-group Correlations

Inter-group correlations are more difficult to cope. When specific features $\varphi_k(X)$ and coupling features of the same task $\varphi_{s,k}(X)$ are strongly correlated, identifiability issues may arise. Indeed, in this case, the predictions might not be injective with relation to the model parameters anymore. This means that deviations in some specific group of parameters $\boldsymbol{\theta}_k$ will be able to be compensated by deviations in the shared parameters $\boldsymbol{\theta}_s$ without impacting the portion of the cost function corresponding to task k. These deviations will then be transmitted through the coupling parameters to the other tasks, leading to a model which might have a good accuracy but whose identified parameters won't make physical sense. An example of this phenomenon is given in Section 8.5.

Dealing with this kind of ill-posedness is uneasy. One of the first attempts to solve this kind of instability in the inverse problems and statistics literature was by the addition of a Tikhonov penalty [Tikhonov, 1943], which corresponds, up to a matrix Γ, to the squared L^2 norm of the parameters: $\|\Gamma\boldsymbol{\theta}\|_2^2$. In most applications, the Tikhonov matrix Γ is equal to the identity matrix I_p. As this penalty is strongly convex, it allows to improve least-squares problems conditioning. Adding it to the block-sparse lasso estimator (17) with some penalty weight $\lambda_t > 0$ leads to

$$\min_{\boldsymbol{\theta}} \sum_{k=1}^{d_x} \left(\sum_{i=1}^{N} \left(Y_{i,k} - \varphi_{s,k}(X_i)^\top \boldsymbol{\theta}_s - \varphi_k(X_i)^\top \boldsymbol{\theta}_k\right)^2 + \lambda_k \|\boldsymbol{\theta}_k\|_1 \right) + \lambda_s \|\boldsymbol{\theta}_s\|_1 + \lambda_t \|\Gamma\boldsymbol{\theta}\|_2^2.$$

In practice, this would correspond to the addition of the same type of penalty in the surrogate lasso problem (20) from proposition 6.1:

$$\min_{\beta} \sum_{i=1}^{N} \|Y_i - B(X_i)\beta\|_2^2 + \lambda_s \|\beta\|_1 + \lambda_t \|\Gamma_\beta \beta\|_2^2, \tag{21}$$

where Γ_β is the scaled Tikhonov matrix

$$\Gamma_\beta = \text{Diag}\left(I_{p_s}, \frac{\lambda_s}{\lambda_1} I_{p_1}, \ldots, \frac{\lambda_s}{\lambda_{d_x}} I_{p_{d_x}}\right) \Gamma. \tag{22}$$

The surrogate lasso hence becomes an *elastic net* [Zou and Hastie, 2005], which interpolates between the lasso and ridge regression [Hoerl, 1962].

The problem that we see with this regularization in our specific case is that it shrinks the model parameters towards 0, which is arbitrary and does not make use of any available knowledge concerning the dynamical system. Instead, we could use the generalized Tikhonov regularization, which consists in the addition of the following penalty

$$\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|_Q^2 := (\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}})^\top Q (\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}), \tag{23}$$

where $\tilde{\boldsymbol{\theta}}$ and Q are supposed to be the expectation and covariance matrix of $\boldsymbol{\theta}$'s prior Gaussian distribution. Despite this original Bayesian interpretation, $\tilde{\boldsymbol{\theta}}$ could also be some prior guess on a subset of the parameters, based on other models or on known orders of magnitude when the parameters have a physical meaning. This can allow not only to improve the problem conditioning, but also to shrink the parameters towards priors that make physical sense. An example of this procedure is presented in Section 8.5.

From a practical perspective, adding the penalty (23) to the block-sparse lasso optimization criterion (17) does not make it more difficult to minimize. Indeed, as summarized in proposition 7.1, the regularized block-sparse lasso problem

$$\begin{aligned} \min_{\boldsymbol{\theta}} \sum_{k=1}^{d_x} \left(\sum_{i=1}^{N} \left(Y_{i,k} - \varphi_{s,k}(X_i)^\top \boldsymbol{\theta}_s - \varphi_k(X_i)^\top \boldsymbol{\theta}_k\right)^2 + \lambda_k \|\boldsymbol{\theta}_k\|_1 \right) \\ + \lambda_s \|\boldsymbol{\theta}_s\|_1 + \lambda_t \|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|_Q^2, \end{aligned} \tag{24}$$

is also equivalent to a surrogate lasso problem. We can hence still solve (24) using the LARS algorithm [Efron et al., 2004], whose computational complexity is equivalent to a matrix inversion. It can also be plugged in the bolasso algorithm from the previous section for its stabilizing effects on the feature selection.

Proposition 7.1 *Given a data set* $\{(X_i, Y_i)\}_{i=1}^N$ *and a sequence of coupling* $(\varphi_{s,k})_{k=1}^{d_x}$ *and specific feature maps* $(\varphi_k)_{k=1}^{d_x}$*, define* $B(X_i)$ *as in* (18)-(19)*, as well as*

$$\tilde{B}(X_i) = \begin{pmatrix} B(X_i) \\ \lambda_t \Gamma_\beta \end{pmatrix}, \quad \tilde{Y}_i = \begin{pmatrix} Y_i \\ \lambda_t \Gamma \tilde{\boldsymbol{\theta}} \end{pmatrix},$$

where Γ *is such that* $Q = \Gamma^\top \Gamma$ *and* Γ_β *is defined as in* (22)*. Consider the following surrogate single-task lasso problem*

$$\min_{\beta} \sum_{i=1}^{N} \|\tilde{Y}_i - \tilde{B}(X_i)\beta\|_2^2 + \lambda_s \|\beta\|_1, \tag{25}$$

and denote its solution by $\hat{\beta} = (\hat{\beta}_s, \hat{\beta}_1, \ldots, \hat{\beta}_{d_x})$*, where* $\hat{\beta}_s$ *corresponds to the first* p_s *components,* $\hat{\beta}_1$ *to the following* p_1 *components and so on. Then, the solution to* (24) *becomes*

$$\hat{\boldsymbol{\theta}}_s = \hat{\beta}_s, \quad \textit{and} \quad \hat{\boldsymbol{\theta}}_k = \frac{\lambda_s}{\lambda_k}\hat{\beta}_k, \quad \textit{for } k = 1, \ldots, d_x.$$

8 Aircraft Dynamics Feature Selection

In this section we apply the system identification techniques from sections 6 and 7 to an aircraft dynamics identification problem.

8.1 Aircraft Dynamics Identification

Aircraft dynamics identification is essential in several engineering application, including for the optimization of flight trajectories. Indeed, historical flight recordings, called *QAR data*, can be used to learn the flight dynamics of an airplane in order to define an approximate optimal control problem (4), as explained in Section 2. We consider hereafter the case where the control problem to be solved aims at finding climb paths which minimize CO_2 emissions and fuel consumption.

Using the notations from table 1, aircraft dynamics during climb can be modeled using the following system of ODE's, derived from Newton's laws of motion and the mass conservation principle:

$$\begin{cases} \dot{h} = & V \sin\gamma, & (26)\\ \dot{V} = & \dfrac{T(\boldsymbol{u},\boldsymbol{x})\cos\alpha - D(\boldsymbol{u},\boldsymbol{x}) - mg\sin\gamma}{m}, & (27)\\ \dot{\gamma} = & \dfrac{T(\boldsymbol{u},\boldsymbol{x})\sin\alpha + L(\boldsymbol{u},\boldsymbol{x}) - mg\cos\gamma}{mV}, & (28)\\ \dot{m} = & -\dfrac{T(\boldsymbol{u},\boldsymbol{x})}{I_{sp}(\boldsymbol{u},\boldsymbol{x})}. & (29) \end{cases}$$

The state and control variables in equations (26)-(29) are respectively $\boldsymbol{x} = (h, V, \gamma, m)$ and $\boldsymbol{u} = (\alpha, N_1)$, supposed to have been measured during previous flights of the same aircraft. The thrust T, the drag D, the lift L and specific impulse I_{sp} are assumed to be unknown functions of the state and control variables. It is quite common in flight mechanics [Hull, 2007, Roux, 2005] to assume that these quantities depend on the following physical variables

$$\begin{cases} T & \text{function of } N_1, \rho, M,\\ D & \text{function of } q, \alpha, M,\\ L & \text{function of } q, \alpha, M,\\ I_{sp} & \text{function of } SAT, h, M, \end{cases} \tag{30}$$

where the air density ρ and static air temperature SAT are assumed to be given (nonlinear) functions of h and the dynamic pressure q and Mach number M are given (nonlinear) functions of the pair (h, V). See for details Appendices A-D. While these four quantities T, D, L, I_{sp} need to be known in order to have a

	Notation	Meaning
States	h	Aircraft altitude
	V	Aircraft true airspeed (TAS)
	γ	Path angle
	m	Aircraft mass
Controls	α	Angle of attack (AOA)
	N_1	Engines turbofan speed
Unknown functions	T	Total thrust force
	D, L	Drag and lift forces
	I_{sp}	Specific impulse
Given models	ρ	Air density
	M	Aircraft Mach number
	SAT	Static air temperature
	q	Dynamic air pressure

Table 1: Variables nomenclature

complete model of the aircraft dynamics, they are typically not measured in flight. They are hence latent features of our dynamical system, which cannot be learned directly. In the next section we propose model structures for these nested quantities.

8.2 Latent Models

A great number of different models of T, D, L and I_{sp} can be found in the literature [Hull, 2007, Roux, 2005] and it can be quite difficult to choose one among them. In this context, it is tempting to look for data-dependent models, whose structure itself is learned during the system identification. It is worth mentioning that this ambition is not specific to aircraft dynamics identification, as many dynamics models come from first-principles involving unmeasurable quantities which have an important physical meaning. While the main ambition of system identification is to accurately predict the future behavior of the system under study, learning models of these latent quantities may be a byproduct of particular interest as well. This is for example the case when identifying the dynamics of chemical reactions, where the concentration of some important chemicals are not measurable.

One characteristic that most T, D, L and I_{sp} models from the literature have in common is that they write as polynomials of the variables listed in (30). More precisely, many models consist in the product between the first variable from the triplets listed and a small polynomial on the remaining pair of variables. This common structure is captured in the following feature map:

$$\begin{array}{ccc} \mathbb{R}^3 & \to & \mathbb{R}^r \\ \Phi_d : (z_1, z_2, z_3) & \mapsto & \left(z_1 \left(z_2^k z_3^{j-k} \right) ; \begin{array}{l} j = 0, \dots, d \\ k = 0, \dots, j \end{array} \right), \end{array}$$

where $d \in \mathbb{N}^*$ is the polynomial degree and $r = \begin{pmatrix} d+2 \\ 2 \end{pmatrix}$. By picking $d > 1$, this last transformation allows to expand our initial triplets of features from (30)

into a higher dimensional space, potentially increasing the explanatory power of our model. For example, for $d = 3$, the drag model features become:

$$\Phi_3(q, \alpha, M) = q(\boldsymbol{\theta}_{D,1} + \alpha\boldsymbol{\theta}_{D,2} + M\boldsymbol{\theta}_{D,3} + \alpha^2\boldsymbol{\theta}_{D,4} + \alpha M\boldsymbol{\theta}_{D,5} + M^2\boldsymbol{\theta}_{D,6} + \alpha^3\boldsymbol{\theta}_{D,7} + \alpha^2 M\boldsymbol{\theta}_{D,8} + \alpha M^2\boldsymbol{\theta}_{D,9} + M^3\boldsymbol{\theta}_{D,10}).$$

Hence, for some $d_T, d_D, d_L, d_{Isp} > 1$, we assume the following structures:

$$\begin{array}{lll} T(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}_T) & = \Phi_{d_T}(N_1, \rho, M) \cdot \boldsymbol{\theta}_T & := X_T \cdot \boldsymbol{\theta}_T, \\ D(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}_D) & = \Phi_{d_D}(q, \alpha, M) \cdot \boldsymbol{\theta}_D & := X_D \cdot \boldsymbol{\theta}_D, \\ L(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}_L) & = \Phi_{d_L}(q, \alpha, M) \cdot \boldsymbol{\theta}_L & := X_L \cdot \boldsymbol{\theta}_L, \\ I_{sp}(\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\theta}_{Isp}) & = \Phi_{d_{Isp}}(SAT, h, M) \cdot \boldsymbol{\theta}_{Isp} & := X_{Isp} \cdot \boldsymbol{\theta}_{Isp}, \end{array} \tag{31}$$

where $\boldsymbol{\theta}_T, \boldsymbol{\theta}_D, \boldsymbol{\theta}_L, \boldsymbol{\theta}_{Isp}$ denote vectors of parameters of sizes p_T, p_D, p_L and p_{Isp}. Note that while models from (31) are polynomials on the triplets of features from (30), they are still linear models on the parameters. Assuming these model structures for the identification of the aircraft dynamics (26)-(29), the feature selection methods described in sections 6 and 7 can be used here in order to build data-depend models of lower complexity and higher accuracy. Furthermore, selecting only a few variables from the feature vectors X_T, X_D, X_L, X_{Isp} would make our model consistent with most of the flight mechanics models, which usually involve only a small number of monomials.

8.3 Multi-task Linear Regression Formulation

Despite the linear nature of the models (31) presented in the previous section, we can see that the dynamical system (26)-(29) remains nonlinear. Hence, the first step for applying the techniques proposed in Section 6 is to arrange these equations as follows:

$$\begin{cases} m\dot{V} + mg\sin\gamma = T(\boldsymbol{u}, \boldsymbol{x})\cos\alpha - D(\boldsymbol{u}, \boldsymbol{x}), & (32) \\ mV\dot{\gamma} + mg\cos\gamma = T(\boldsymbol{u}, \boldsymbol{x})\sin\alpha + L(\boldsymbol{u}, \boldsymbol{x}), & (33) \\ 0 = T(\boldsymbol{u}, \boldsymbol{x}) + \dot{m}I_{sp}(\boldsymbol{u}, \boldsymbol{x}). & (34) \end{cases}$$

These transformations are of the same nature as in the abstract example (10)-(11). Note that equation (26) was dropped as it does not contain any parameter to be identified. We see that the obtained system (32)-(34) is now linear on the parameters, as required by the block-sparse lasso. Using (31), these dynamics can be cast as a set of linear regression models, in the spirit of the *equation-error method*:

$$\begin{cases} Y_1 = X_{T1} \cdot \boldsymbol{\theta}_T - X_D \cdot \boldsymbol{\theta}_D + \varepsilon_1, & (35) \\ Y_2 = X_{T2} \cdot \boldsymbol{\theta}_T + X_L \cdot \boldsymbol{\theta}_L + \varepsilon_2, & (36) \\ 0 = X_T \cdot \boldsymbol{\theta}_T + X_{Ispm} \cdot \boldsymbol{\theta}_{Isp} + \varepsilon_3, & (37) \end{cases}$$

where $\varepsilon_1, \varepsilon_2, \varepsilon_3$, are random errors of mean 0 and

$$\begin{array}{c} Y_1 = m\dot{V} + mg\sin\gamma, \quad Y_2 = mV\dot{\gamma} + mg\cos\gamma, \\ X_{T1} = X_T\cos\alpha, \quad X_{T2} = X_T\sin\alpha, \quad X_{Ispm} = \dot{m}X_{Isp}. \end{array} \tag{38}$$

It is clear from (35)-(37) that, as in (15), we are in the scope of structured dynamics. Indeed, the thrust parameters $\boldsymbol{\theta}_T$ are shared across all three equations,

while $\boldsymbol{\theta}_D, \boldsymbol{\theta}_L, \boldsymbol{\theta}_{Isp}$ are task-specific. In this setting, X_T, X_{T1} and X_{T2} play the role of the coupling features $(\varphi_{s,k}(X))_{k=1}^{d_x}$ and $-X_D, X_L, X_{Ispm}$ are the specific features $(\varphi_k(X))_{k=1}^{d_x}$, where

$$X = (\boldsymbol{u}, \boldsymbol{x}, \dot{\boldsymbol{x}}) = (\alpha, N_1, h, V, \gamma, m, \dot{h}, \dot{V}, \dot{\gamma}, \dot{m}), \tag{39}$$

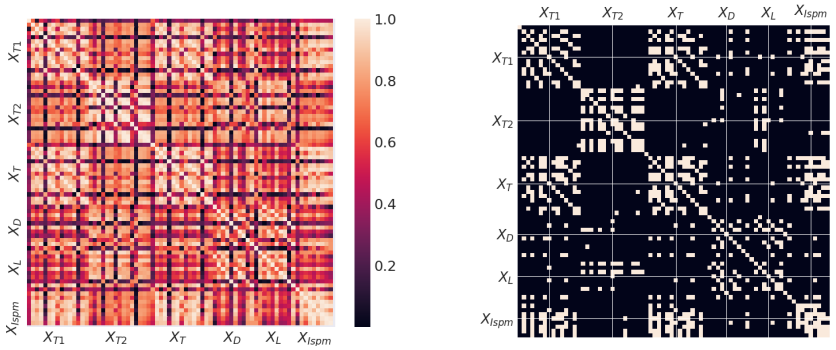
is assumed to be random, as well as $Y = (Y_1, Y_2, 0)$. We suppose the availability of a training set of N i.i.d observations $\{(X_i, Y_i)\}_{i=1}^N$ derived from states and controls measurements through (38)-(39). We are hence in the scope of application of the block-sparse lasso (17). In this setting, the block-sparse feature matrix (19) will write

$$B(X_i) = \begin{pmatrix} X_{T1}^\top & -X_D^\top & 0 & 0 \\ X_{T2}^\top & 0 & X_L^\top & 0 \\ X_T^\top & 0 & 0 & X_{Ispm}^\top \end{pmatrix} \in \mathbb{R}^{3 \times p}, \tag{40}$$

where $p = p_T + p_D + p_L + p_{Isp}$.

8.4 Feature Selection Stabilization with the Bolasso

As commonly experienced in polynomial regression, the groups of features X_{T1}, X_{T2}, X_T, X_D, X_L and X_{Ispm} are highly correlated in practice. This is visible on Figure 1, where the diagonal blocks correspond to intra-group correlations, all other cells being inter-group correlations. The right-panel shows that intra-



(a) (b)

Figure 1: (a) Absolute value of the correlations between features from (35)-(37). (b) Highlight in white of the absolute correlations greater than 0.9 .

group correlations are quite high (most are superior than 0.9 in absolute value), which makes the block-sparse lasso selection inconsistent, as explained in Section 7. This can be stabilized using the bolasso (algorithm 1).

We have good reasons to believe that the assumptions presented in 7.1 on which this algorithm's consistency relies are verified in our setting. Indeed, assumption **(H1)** is equivalent to requiring finite cumulant generating functions for Y_1, Y_2, X_ℓ, $\ell \in \{T1, T2, T, D, L, I_{spm}\}$. This is the case when $\varepsilon_1, \varepsilon_2, X_\ell$, have compact support, which is verified in our case due to the physical nature of our

data. Assumption **(H2)** summarizes here to each non-zero block of X having invertible second order moments, which has to be verified for our model to be identifiable. Assumption **(H3)** is of course equivalent to assuming the existence of the multi-task regression model (35)-(37). Numerical results supporting this claim are presented in Section 9.

8.5 Taylored Generalized Tikhonov Regularization

Given the estimated parameters $\hat{\boldsymbol{\theta}}$ obtained by applying algorithm 1, one can use equations (31) to have access to predictions of the latent functions $\hat{T}, \hat{D}, \hat{L}$ and $\hat{I}_{sp}$. In practice, one may observe that every execution of the *block sparse lasso* (17) leads to $\hat{T}$ and $\hat{I}_{sp}$ systematically equal to 0, all their parameters being rejected by the feature selection procedure. This seems to occur because we are regressing a function constantly equal to 0 in the last task (37), whose trivial solution is indeed setting all parameters in $\boldsymbol{\theta}_T$ and $\boldsymbol{\theta}_{Isp}$ to 0. The other equations (35)-(36) from our multi-task model should prevent this from happening, as they also share $\boldsymbol{\theta}_T$. This is however not the case here because our model is corrupted by strong inter-group correlations, as evidenced by the white cells outside the diagonal blocks on Figure 1. We see indeed that two groups of coupling features, X_{T1} and X_T, are highly correlated to each other, because α in (38) is small. We also see that these features present absolute correlations greater than 0.9 with some components of all the groups of specific features X_D, X_L and X_{Ispm}. Hence, it is possible to find solutions to problem (35)-(37) where targets Y_1 and Y_2 are completely explained by D and L, their parameters having compensated the absence of T. This is coherent with the fact that, even when the L^1 penalty is omitted from (17) and plain least-squares is applied, the predicted thrust and specific impulse $\hat{T}, \hat{I}_{sp}$ obtained are positive but really small compared to the known orders of magnitude of these physical quantities, as shown later on Figure 5.

In order to predict forces with the good order of magnitude, while still keeping a good accuracy in terms of states derivatives, we tried using the *regularized block-sparse lasso* (24). For this, we assume the availability of one or more prior estimators of some of the latent functions T, D, L or I_{sp}. A simple prior estimator could be in this case a constant function equal to the known order of magnitude of given quantity. Without loss of generality, we suppose in what follows that $\tilde{I}_{sp}$ is such a prior estimator of I_{sp}. In this case, setting the Tikhonov matrix as follows

$$\Gamma\tilde{\boldsymbol{\theta}} := \tilde{I}_{sp}, \quad \Gamma := (\underbrace{0, \ldots, 0,}_{p_T+p_D+p_L} X_{Isp}^\top),$$

brings the generalized Tikhonov penalty from (23) to be

$$\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|_Q^2 = \|\Gamma(\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}})\|_2^2 = \|X_{Isp}^\top \boldsymbol{\theta}_{Isp} - \tilde{I}_{sp}\|_2^2. \tag{41}$$

We see that, by construction, this regularization favors solutions whose I_{sp} predictions do not deviate to much from the prior. Because of the existing coupling between the tasks of our regression model (35)-(37), this additional penalty should suffice to bring all latent quantities to the correct orders of magnitude. This is illustrated by the results presented in the following section.

Remark We also tried to solve this identification problem by using an Alternate Least-Squares scheme [De Leeuw et al., 1976] in order to separate the estimation

of the groups of parameters of T and (D, L, I_{sp}). This strategy did not work well in practice as it still converged slowly to the trivial solution $\hat{T} = \hat{I}_{sp} = 0$. For this reason, we decided not to present this approach in this paper.

9 Experiments

This section summarizes the protocols and results of experiments with real flight data designed to assess the block-sparse bolasso in the context of aircraft dynamics identification.

9.1 Data Set Description and Preprocessing

QAR data description We present in this section numerical experiments carried using real flight data, extracted from Quick Access Recorder devices (QAR) of 25 medium haul aircraft of the same type. Although the original data set contains raw measurements of thousands of different variables, only the following were used herein: $(h, M, C, N_1, SAT, \Theta)$. The reader is referred to table 1 for the notations. Only data concerning the climb phase of the flights were kept, i.e. data corresponding to altitudes between FL50 = 5 000 ft and the *top of climb* (cruise altitude), specific to each flight. This is due to the fact that the estimated dynamics are used for the optimization of the climb profiles of these aircraft. The obtained data set contains 8 261 619 observations, made of 10 471 different flights sampled at 1 measurement per second.

Derivation of state and control observations As classically done in *collocation methods* (explained at the end of Section 3), the raw QAR signals were smoothed using univariate smoothing splines, which allow to compute their derivatives. They were then used to derive all the state and control variables $\boldsymbol{x}, \boldsymbol{u}$ from (39), through standard flight mechanics formulas presented in the Appendices A-D. Observations of state derivatives $\dot{\boldsymbol{x}}$ were then computed analytically, based on the splines-derivatives of the smoothed signals. This constitutes the main use of the splines preprocessing step, as no measurements of the states derivatives are available. Finally, the degrees of the polynomial models (31) of T, D, L and I_{sp} were set to $d_T = 4$ and $d_D = d_L = d_{Isp} = 3$. This corresponds to $p_T = 15$ features for the thrust and $p_D = p_L = p_{Isp} = 10$ features for the other latent quantities.

9.2 Feature Selection Assessment

In order to assess the quality of the feature selection performed by the *block-sparse bolasso* presented in sections 7.1 and 8.4, algorithm 1 was run separately on the datasets of each different aircraft, and the obtained sparsity patterns were compared to each other. The Tikhonov regularization parameter λ_t introduced in sections 7.2 and 8.5 was set to 1000 for this part of the experiments, which is justified later in Section 9.3. The prior model $\tilde{I}_{sp}$ used here to define such regularization was taken from Roux [2005]. After scaling the features and targets, we chose to set the multiple L^1 penalty parameters of the *regularized block-sparse lasso* (24) to be equal: $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_s$. The common parameter was then set using 30-fold cross-validation. This was performed separately

for each data set, on 33%-validation sets, and done a single time, prior to the replications of the bolasso. For all experiments, the number of bootstrap replications was set to $b = 128$. Solving the multiple surrogate lasso problems (25) was done using the least angle regression algorithm (LARS), implemented in Python's SCIKIT-LEARN.LINEAR_MODEL library [Pedregosa et al., 2011].

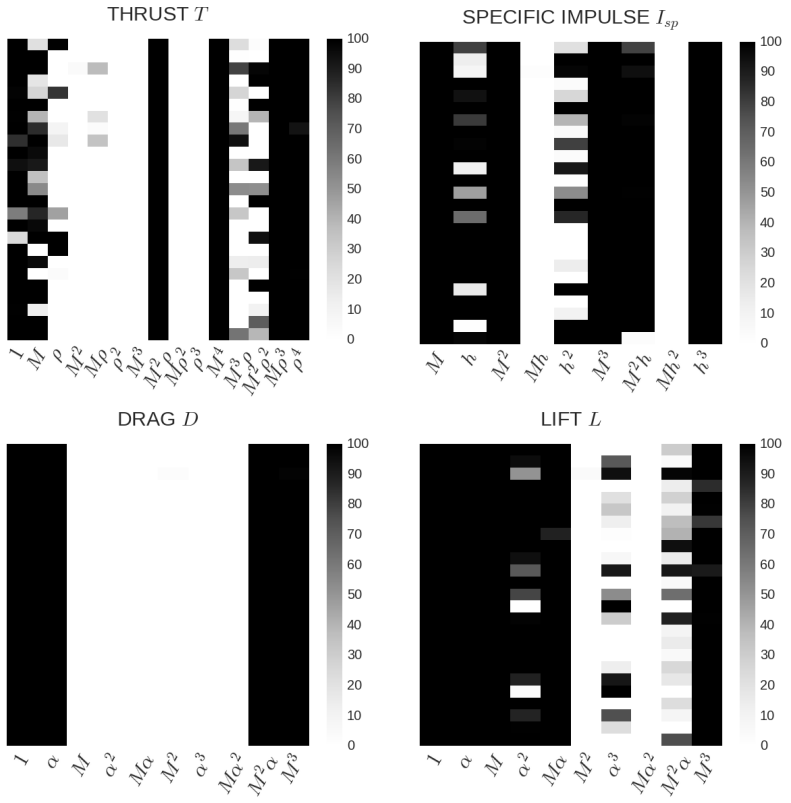


Figure 2: Feature selection results for the T, D, L and Isp models. The columns correspond to possible features for T/N_1, D/q, L/q and I_{sp}/SAT.

Results The results of the feature selection performed on the 25 data sets are displayed on Figure 2. Each row of these matrices corresponds to a different aircraft and each column to a different feature. The cells colors encode the frequency of selection of given feature for given aircraft across all the block-sparse lasso executions. It can be observed that most dark columns are quite homogeneous, which indicates that similar features were selected for the majority of aircraft. This seems to validate our approach for this kind of data, as one would expect that airplanes of the same type should have physical models for the thrust, drag, lift and specific impulse with similar structures. As an example, a common sparse model for the thrust force T would be here made of the features: $X_T = N_1(1, \rho M^2, M^4, \rho^2 M^2, \rho^3 M, \rho^4)$. These results show that the feature selection method allows to keep only 21 features of the 45 initial candidates, which represents a 53% compression.

9.3 Dynamics Estimation Assessment

The quality of the *regularized block-sparse bolasso* as an aircraft dynamics estimator was assessed using a subset of $m = 424$ flights, corresponding to a single aircraft and comprising 334 531 observations.

For this, our model was trained on $m - 1$ flights, leaving out one (randomly chosen) flight for testing. Let n denote the number of observations from the test flight $\{(\boldsymbol{u}_{test}(t_i), \boldsymbol{x}_{test}(t_i), \dot{\boldsymbol{x}}_{test}(t_i))\}_{i=1}^{n}$, whose state derivatives were derived as explained in Section 9.1. A first quality assessment strategy was to compare the observed states derivatives of the test flight $\dot{\boldsymbol{x}}_{test}(t_i)$ to the predictions of the dynamics model $g(\boldsymbol{x}_{test}(t_i), \boldsymbol{u}_{test}(t_i), \hat{\boldsymbol{\theta}})$, $i = 1, \ldots, n$. We also compared the obtained estimations to the predictions of *multi-task nonlinear least-squares* (*NLLS*), proposed in Rommel et al. [2017]. This competing approach, which also falls in the category of *equation-error methods*, does not consider linearizing the dynamics with relation to the parameters (as explained in Section 5), leading to a regression function G which is nonlinear due to the mass dynamics (29). It then consists in directly minimizing the squared-error of regression model (7) using Levenberg-Marquardt algorithm [Levenberg, 1944, Marquardt, 1963]:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{d_x} \sum_{i=1}^{N} (Y_{i,k} - G_k(X_i, \boldsymbol{\theta}))^2, \tag{42}$$

where G_k denotes the k^{th} component of function G. Note that, unlike the block-sparse bolasso, this nonlinear least-squares approach requires an initial guess for the parameters as the training is done through an iterative optimization algorithm. In the following experiments, the pretrained specific consumption model from Roux [2005] was once again used to set the initial parameters of I_{sp}, while the initial parameters of T, D and L were set using unpenalized single-task ordinary least-squares. We refer the reader to Rommel et al. [2017] for more details on this matter.

Qualitative results for one representative flight are shown on Figure 5, where three different values of Tikhonov penalty parameter λ_t were considered. In the three bottom panels, we see that the state derivatives estimated by all four algorithms lie very close to each other, and match reasonably well to the recorded flight. The four top panels show however that the predicted thrust T, drag D, lift L and specific impulse I_{sp}, which are hidden in the dynamics models, are really different from one algorithm to the other. Namely, the nonlinear least-squares leads to predicted thrust, drag and specific impulse which are too high compared to the correct orders of magnitude, while insufficiently regularized block-sparse bolasso (for $\lambda_t = 10$ and 100) predicts negative drag, which is physically incorrect.

The mean-squared errors of *NLLS* and regularized block-sparse bolasso (*BSBL*) with $\lambda_t = 1000$ in terms of predicted states derivatives were computed and cross-validated over all flights. The histograms of the errors obtained are depicted on Figure 3, as well as their medians. Both mean error patterns are really close to each other and the difference between their median values are not statistically significant. The only variable which seems to be predicted slightly less accurately by the block-sparse bolasso than nonlinear least-squares is the speed derivative. These results confirm that the regularized block-sparse bolasso has similar accuracy to multi-task nonlinear least-squares for aircraft dynamics es-

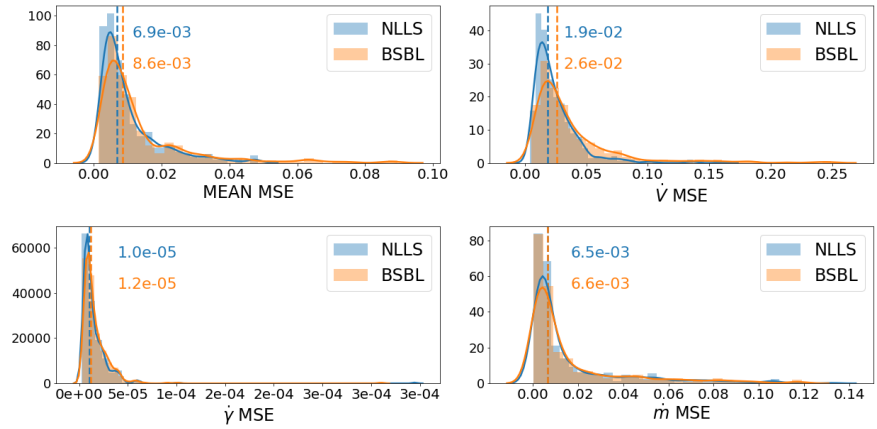


Figure 3: Leave-one-out off-sample errors distributions for nonlinear least-squares *NLLS* and block-sparse bolasso *BSBL*. Median errors are annotated and marked by dashed vertical lines.

timation, with 53% fewer features, a better physical consistency of its latent functions T, D, L and I_{sp} and without requiring an initial guess for the model parameters.

9.4 Flight Resimulation

The issue with the assessment strategy from Section 9.3 is that it is static: it does not incorporate the fact that the observations are time dependent, nor does it take into account the goal of optimally controlling the aircraft system. Another quality criterion more aligned with this considerations is to compare the observed states and controls of the test flight $(\boldsymbol{x}_{test}(t_i), \boldsymbol{u}_{test}(t_i))$ to the solution of the following optimal control problem

$$\begin{aligned} \min_{(\boldsymbol{x},\boldsymbol{u})} \int_{t_0}^{t_n} \left(\|\boldsymbol{u}(t) - \boldsymbol{u}_{test}(t)\|_{\boldsymbol{u}}^2 + \|\boldsymbol{x}(t) - \boldsymbol{x}_{test}(t)\|_{\boldsymbol{x}}^2 \right) dt \\ \text{s.t.} \quad \dot{\boldsymbol{x}}(t) = g(\boldsymbol{x}(t), \boldsymbol{u}(t), \hat{\boldsymbol{\theta}}), \end{aligned} \tag{43}$$

where $\|\cdot\|_{\boldsymbol{u}}$, $\|\cdot\|_{\boldsymbol{x}}$ denote scaling norms. The main idea of problem (43) is to try to simulate the observed test flight using the learned dynamics $g(\cdot,\cdot,\hat{\boldsymbol{\theta}})$. For this, we seek controls similar to the test flight's controls $\boldsymbol{u}_{test}$ which lead to states as close as possible to the observed trajectory $\boldsymbol{x}_{test}$. In practice, problem (43) was solved using the optimal control solver BOCOP [Bonnans et al., 2017], which implements the *direct-transcription method* [Betts, 2010] and uses the interior point solver IPOPT [Wächter and Biegler, 2006].

A comparison between a real test flight and the solutions of (43) using *NLLS* and regularized block-sparse bolasso as dynamics estimators are presented on Figure 6. Simulated states and controls using both methods seem to be relatively good approximations of the recorded flight. This is confirmed in the first frame of Figure 4, which shows the histogram of mean squared simulation

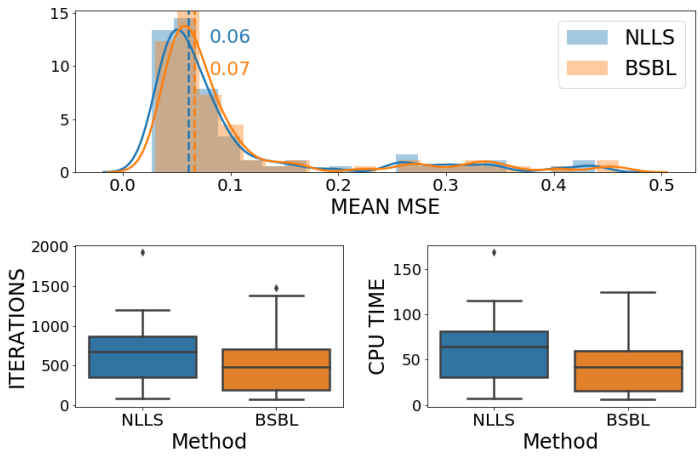


Figure 4: Distribution of the off-sample simulation error and boxplot of the optimization number of iterations and CPU time.

errors for both methods, as well as the median error (vertical dashed-lines). The bottom frames represent the number of iterations and CPU time needed by the interior point algorithm used to solve the simulation problem (43). We see that, while the error distributions are really similar for both methods, the computational cost of the block-sparse bolasso in terms of iterations and time is more variable than multi-task nonlinear least-squares but its median value is about 30% lower. This is encouraging as these results suggest that block-sparse bolasso method allows to obtain surrogate optimal control problems (4) as close to the original problems (2) as multi-task nonlinear least-squares, at a fairly reduced optimization cost.

10 Concluding Remarks

In this paper we focused on the identification of structured nonlinear systems for model-based reinforcement learning purposes. While traditional methods in this framework are based on numerically solving several ODE's and make use of unregularized maximum likelihood parameter estimation, we propose a novel approach which extends to this setting well-known supervised learning methods such as the lasso, ridge regression and bootstrap stabilization. Our estimator, coined regularized block-sparse bolasso, is proven to achieve consistent feature selection, while preserving the dynamical system's structure and leveraging the existing couplings between differential equations. Moreover, we also showed that the developped statistical models can be efficiently trained, as they can always be converted into a series of surrogate lasso problems, which are solvable at the complexity cost of a matrix inversion using the modified LARS implementation. In order to verify the usefulness and performance of our approaches in real world problems, we applied our method to an aircraft trajectory optimization problem and compared it to other well-established solutions. The numerical results

show that the regularized block-sparse lasso is more scalable than traditional output-error methods. Although no improvement in dynamics prediction accuracy is observed compared to standard unregularized equation-error approaches, this new technique does not require any initialization and is shown to lead to sparse descriptions which are flexible, light and interpretation-friendly. The results also illustrate how the use of generalized Tikhonov regularization leads to dynamics estimators with internal hidden components (aerodynamic forces in our example) agreeing with physical knowledge, which was not guaranteed by previous approaches.

From a nonlinear system identification perspective, our work tries to answer several open problems of this field pointed out by Ljung [2010], such as automatically finding useful parametrizations without loosing interpretability, having general ways of using structural problem specific insight and adapting system identification to contexts of large amounts of data. We hope that our contribution helps fostering new interactions between the control and the machine learning community, encouraging further extensions of existing techniques from one field to the other, which should help to solve more and more challenging continuous control problems with model-based approaches.

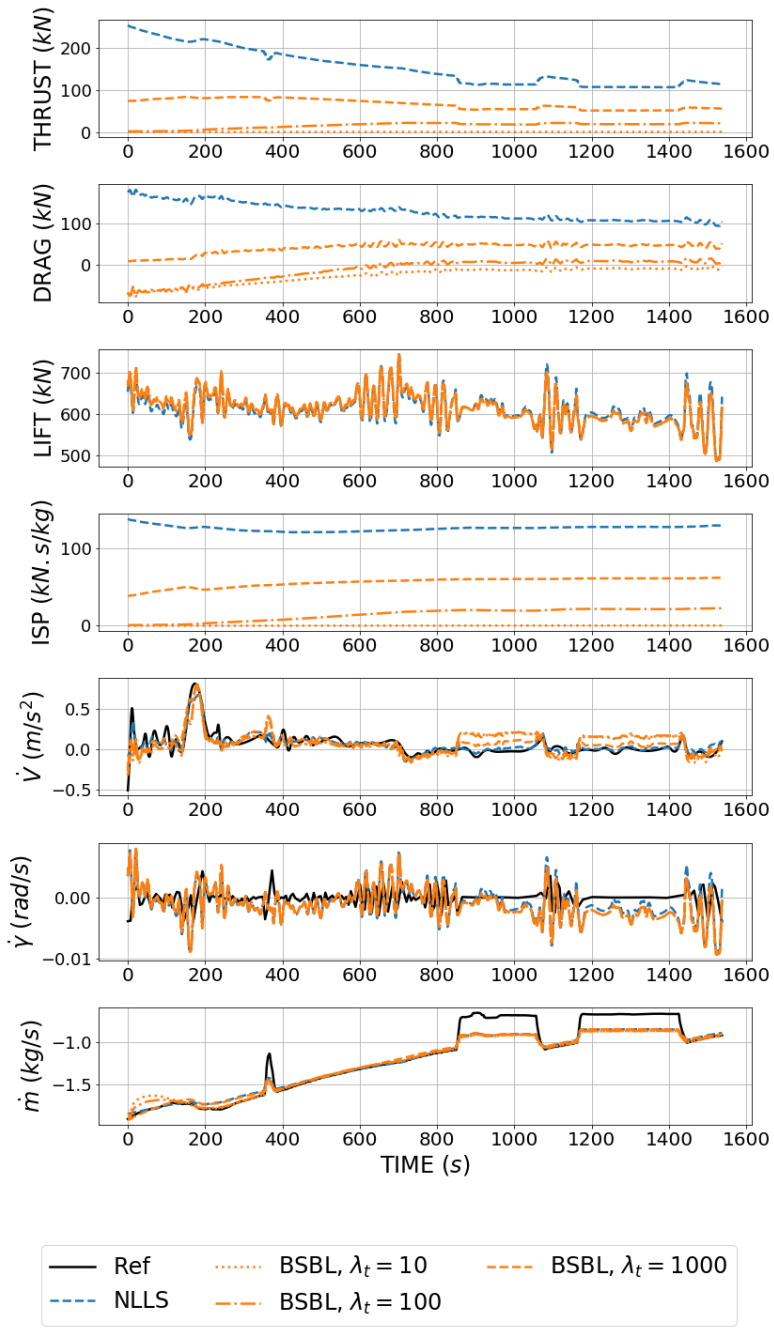


Figure 5: Predicted dynamics using multi-task nonlinear least-squares (*NLLS*) and Block-sparse bolasso model (*BLBS*) with different regularization parameters λ_t.

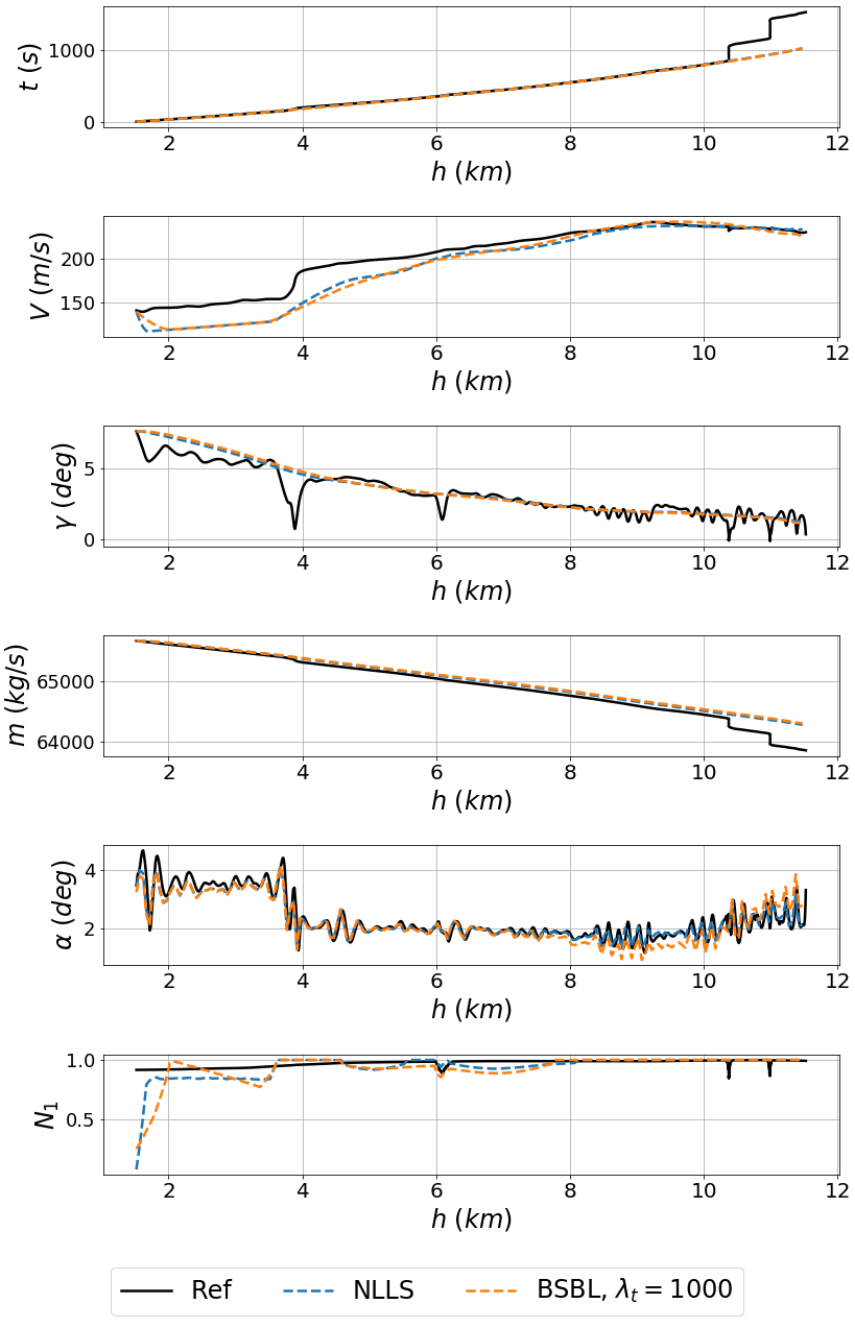


Figure 6: Resimulated trajectory using multi-task nonlinear least-squares (*NLLS*) and Block-sparse bolasso model (*BLBS*).

Appendices

A Air density

The International Standard Atmosphere model gives that $\rho = \frac{P}{R_s SAT}$, where P is the atmospheric pressure expressed in Pascals, SAT is the *Static Air Temperature* in Kelvins and $R_s = 287.053 J.kg^{-1}.K^{-1}$.

B Static Air Temperature

The SAT involved in the air density model may be expressed as a function of the altitude h in meters, which is a state variable and a measured variable available through QAR data:

$$SAT(h) = T_0 + \alpha_T h,$$

with $\alpha_T = -0,0065 K/m$ and $T_0 = 288,15 K$. This last expression is valid if we stay in the troposphere ($h \leq 11000 m$). Another more accurate model exist, which can be computed using QAR data:

$$SAT(TAT, M) = \frac{TAT}{1 + \frac{\lambda - 1}{2} M^2}, \tag{44}$$

where $\lambda = 1.4$ and SAT, TAT are expressed in Kelvins. We say (44) is more accurate because it only uses in-flight data and does not depend on the temperature at sea level, T_0, which should vary with the geographic position and time of the flight.

C TAS, Mach number and sound speed

The Mach number is a function of the SAT and the aircraft relative speed in meters per second V, also called *True Airspeed* (TAS):

$$M = \frac{V}{V_{sound}} = \frac{V}{(\lambda R_s SAT)^{\frac{1}{2}}},$$

V_{sound} being the atmospheric sound speed in meters per second. Consequently, M can either be seen as a measured variable available in QAR data or as a function of two state variables h and V.

D Flight mechanics model

The path angle γ is the angle between the aircraft speed vector and the horizontal direction. The angle of attack α is the angle between the wings' chord and the relative wind. Here we assume the wings' chord is aligned with the thrust vector and with the aircraft longitudinal axis. The pitch Θ is the angle between the longitudinal axis and the horizontal axis. Such definitions and assumptions lead to the following equation linking these three variables: $\Theta = \alpha + \gamma$.

References

R. M. Anderson and R. M. May. *Infectious Diseases of Humans: Dynamics and Control*. Oxford university press, 1992.

A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 2008.

F. Bach. Bolasso: model consistent Lasso estimation through the bootstrap. In *Proceedings of the 25th International Conference on Machine Learning*, pages 33–40, 2008.

J. T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. SIAM, 2010.

J. F. Bonnans, D. Giorgi, V. Grelard, B. Heymann, S. Maindrault, P. Martinon, O. Tissot, and J. Liu. Bocop – A collection of examples. Technical report, INRIA, 2017. URL `http://www.bocop.org`. http://www.bocop.org/.

R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

J. De Leeuw, F. W. Young, and Y. Takane. Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41(4):471–503, 1976.

S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu. On the sample complexity of the linear quadratic regulator. *arXiv preprint arXiv:1710.01688*, 2017.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32:407–499, 2004.

T. Evgeniou, C. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *Journal Machine Learning Research*, 6:615–637, 2005.

R. FitzHugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal*, 1(6):445–466, 1961.

J. Friedman, T. Hastie, and R. Tibshirani. A note on the Group Lasso and a Sparse group Lasso. arXiv:1001.0736, 2010.

A. E. Hoerl. Application of ridge analysis to regression problems. *Chemical Engineering Progress*, 58:54–59, 1962.

D. G. Hull. *Fundamentals of Airplane Flight Mechanics*. Springer, 2007.

R. V. Jategaonkar. *Flight Vehicle System Identification: A Time Domain Methdology*. AIAA, 2006.

V. Klein and E. A. Morelli. *Aircraft System Identification*. AIAA, 2006.

B. Krishnapuram, L. Carin, M. A. Figueiredo, and A. J. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):957–968, 2005.

K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

L. Ljung. *System Identification: Theory for the User*. Prentice-hall, 1987.

L. Ljung. Perspectives on system identification. *Annual Reviews in Control*, 34 (1):1–12, 2010.

L. Ljung and S. T. Glad. On global identifiability for arbitrary model parameterizations. *Automatica*, 30(2):265–276, 1994.

J. Lokhorst. The lasso and generalised linear models. *Honors Project, The University of Adelaide, Australia*, 1999.

R. E. Maine and K. W. Iliff. *Identification of Dynamic Systems: Theory and Formulation*. NASA, STIB, 1985.

D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11 (2):431–441, 1963.

N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.

J. Nagumo, S. Arimoto, and S. Yoshizawa. An active pulse transmission line simulating nerve axon. *Proceedings of the IRE*, 50(10):2061–2070, 1962.

G. Obozinski, B. Taskar, and M. I. Jordan. Multi-task feature selection. In *ICML-06 Workshop on Structural Knowledge Transfer for Machine Learning*, 2006.

F. Pedregosa et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

N. K. Peyada and A. K. Ghosh. Aircraft parameter estimation using a new filtering technique based upon a neural network and Gauss-Newton method. *The Aeronautical Journal*, 113(1142):243–252, 2009.

N. K. Peyada, A. Sen, and A. K. Ghosh. Aerodynamic characterization of hansa-3 aircraft using equation error, maximum likelihood and filter error methods. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 2008.

J. O. Ramsay, G. Hooker, D. Campbell, and J. Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007.

B. Recht. A tour of reinforcement learning: The view from continuous control. *arXiv preprint arXiv:1806.09460*, 2018.

J. F. Ritt. *Differential Algebra*, volume 33. American Mathematical Soc., 1950.

C. Rommel, J. F. Bonnans, B. Gregorutti, and P. Martinon. Aircraft dynamics identification for optimal control. In *Proceedings of the 7th European Conference for Aeronautics and Aerospace Sciences*, 2017.

E. Roux. *Pour une approche analytique de la dynamique du vol.* PhD thesis, Supaero, 2005. URL `http://elodieroux.com/ReportFiles/TheseElodie.pdf`.

D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354, 2017.

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 58:267–288, 1994.

R. Tibshirani. The lasso method for variable selection in the Cox model. *Statistics in medicine*, 16(4):385–395, 1997.

A. N. Tikhonov. On the stability of inverse problems. In *Doklady Akademii Nauk SSSR*, volume 39, pages 195–198, 1943.

S. van de Geer. $\ell 1$-regularization in high-dimensional statistical models. In *Proceedings of the International Congress of Mathematicians 2010 (ICM 2010)*, volume 4, pages 2351–2369, 2010.

J. M. Varah. A spline least squares method for numerical parameter estimation in differential equations. *SIAM Journal on Scientific and Statistical Computing*, 3(1):28–46, 1982.

A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68:49–67, 2005.

P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7(Nov):2541–2563, 2006.

H. Zou. The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association*, 101:1418–1429, 2006.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.